Development of Communication System between TPMS and Server using Combination of OFDM and Convolutional Code Technique Based on SDR

Hendy Briantoro[1]*, Billy Montolalu[2], Ardiansyah Al Farouq[3]
Department of Computer Engineering, School of Electrical Engineering [1], [3]
Department of Software Engineering, School of Informatics [2]
Telkom University, Surabaya Campus
Surabaya, Indonesia
hendybr@telkomuniversity.ac.id[1], billymontolalu@telkomuniversity.ac.id [2], farouq@telkomuniversity.ac.id[3]

Abstract— **The Tire Pressure Monitoring System (TPMS) has evolved into an essential element of contemporary vehicles, playing a pivotal role in enhancing road safety and the overall driving experience. Traditionally, TPMS systems rely on dedicated hardware components for the collection and transmission of tire pressure data to the vehicle's onboard computer and the data is visible only to the driver. In this research, we have developed a wireless communication system between TPMS and a server, enabling tire pressure data to be accessible not only to the driver but also remotely traceable by others. To build a reliable communication system, we utilized a combination of Orthogonal Frequency Division Multiplexing (OFDM) and Convolutional Code technologies. This system is implemented using Software-Defined Radio (SDR) technology. This communication method employs OFDM to enhance data throughput and integrates Convolutional Code to mitigate errors in received data. Consequently, this approach achieves a maximum throughput of 119.19MBps when utilizing the OFDM system alongside 16QAM modulation. The bit error rate for received data without coding stands at 5.77%, but the application of Convolutional Code with a 1/2 code rate effectively reduces this error rate to 3.85%. This system improves the reliability of TPMS communication with the server while also ensuring a consistently high throughput. It enhances road safety and remote monitoring capabilities.**

Keywords— tire pressure monitoring system, software defined radio, OFDM, convolutional code

I. Introduction

The automotive industry has witnessed tremendous progress in recent years, with an increasing emphasis on improving vehicle safety and performance. An important aspect of vehicle safety is monitoring and maintaining tire pressure, as improper tire pressure can lead to reduced fuel efficiency, impaired handling and increased risk of accidents. In this context, the Tire Pressure Monitoring System (TPMS) has become an integral component of modern vehicles, making a significant contribution to road safety and the overall driving experience [1][2].

Traditionally, TPMS rely on specialized hardware components to collect and transmit tire pressure data to the onboard computer within the vehicle. The tire pressure data can only be seen by the driver of the vehicle. Now, tire pressure data shouldn't just be accessible to the driver but also trackable remotely by other individuals or entities such as family members, companies, or colleagues. This aims to enable these parties to notify the driver if the tire pressure exceeds or falls below the standard limit. This data can be stored not only within the vehicle but also on a server situated remotely from the vehicle.

In [3], the authors the author developed a TPMS that presents tire pressure information on a dot matrix Liquid-Crystal Display (LCD). The pressure data is transmitted via a radio frequency (RF) module operating at 433MHz. Similarly, paper [4] explored wireless communication between tire sensors and TPMS modules. In this study, the author analyzes the power consumption associated with this communication through simulation. In [5], the authors conducted experiments involving the transmission of data from the tire sensor to the TPMS module via wireless communication operating at a frequency of 433MHz. Additionally, the paper examines the transmission of data from the TPMS module to a computer using wired communication employing the universal synchronous/asynchronous receiver/transmitter (USART) protocol.

Nonetheless, the latest progress in wireless communication technologies, in conjunction with the application of Software-Defined Radio (SDR) and Orthogonal Frequency Division Multiplexing (OFDM), has ushered in a new era for redefining the way TPMS communicates with servers [6]. This innovative combination of SDR and OFDM technology holds the potential to provide more robust, efficient, and reliable communication between the TPMS and centralized server systems.

OFDM is a bandwidth-efficient multicarrier digital communication scheme in which OFDM subcarriers overlap, resulting in spectrum efficiency [7][8]. OFDM offers advantages in increasing data transmission rates compared to single-carrier communication systems. In OFDM, signals are divided into multiple subcarriers, leading to faster data transmission. Research on OFDM has been previously conducted by [9]. Their research concentrated on assessing the performance of OFDM with the utilization of SDR in a real-

p-ISSN 2301-7988, e-ISSN 2581-0588
DOI : 10.32736/sisfokom.v13i2.2024, Copyright ©2024
Submitted : November 29, 2023, Revised : April 16, 2024 , Accepted : April 25, 2024, Published : June 5, 2024

time scenario. This study was conducted through the application of two approaches: OFDM implemented in simulation using GNU Radio and practical implementation of OFDM [10].

SDR is a technology where the hardware remains constant but allows for changes on the software side. This enhances flexibility in the choice of wireless communication technologies and improves hardware utilization. One of the hardware platforms that supports SDR is Universal Software Peripheral Radio (USRP) [8]. Research on communication technology using USRP has been conducted before, including studies on data transmission using various modulation types with USRP [9].

The primary distinction between this study and earlier ones lies in the methodology and implementation approach. In prior research conducted by Gayatri Phade, Aniket Kulkarni, Omkar Vaidya, and Sanjay Gandhe, they devised a TPMS where pressure data was transmitted using a radio frequency (RF) module operating at 433MHz. However, in their setup, tire pressure information was solely showcased on a local display without being sent to a server. Suk-seung Hwang, Seong-min Kim, Jae-Young Pyun, and Goo-Rak Kwon explored wireless communication between tire sensors and TPMS modules, but their evaluation was conducted through simulation. Slawomir Grys developed a communication system between TPMS modules and computers, but it relied on wired communication. The current research developed a wireless communication system that leverages OFDM and Convolutional Code to establish seamless communication between TPMS and server. This system is implemented using SDR technology. SDR enables the dynamic reconfiguration of radio parameters, making it a versatile and adaptable platform for various wireless communication applications. OFDM offers the advantage of robustness in high-noise environments and efficient data transmission through its use of orthogonal subcarriers. Convolutional code serves the purpose of minimizing errors during data transmission.

II. METHODOLOGY AND SYSTEM DESIGN

In this section, we explain the methodology and system design in the paper.

A. Methodology

Tire Pressure Monitoring System (TPMS) is system can monitor the air pressure inside the tires on vehicles. TPMS can be categorized into two primary groups: direct systems and indirect systems. In direct systems, an individual sensor is installed on each vehicle tire, allowing for precise measurement of tire pressure. In contrast, indirect systems use sensors onboard to estimate tire pressure. Both systems detect tire pressure changes by closely observing variations in measured values compared to the standard tire pressure [11][12].

The direct TPMS offers real-time and accurate pressure information for each tire, while the indirect TPMS utilizes existing onboard sensors to indirectly determine tire pressure, offering a more budget-friendly option. These monitoring systems significantly enhance vehicle safety by notifying drivers about potential tire pressure deviations, thus helping to prevent risky situations and improving overall road safety [13].

Software-Defined Radio (SDR) is a radio communication system in which blocks of radio communication such as filters, amplifiers, and modulators are implemented on a computer or other embedded devices like FPGA. By using SDR, it's possible to save on hardware costs because when there are changes to a block in the communication system, the hardware being used doesn't need to be altered—only the software side needs to be changed [14]. There are several software tools that support the implementation of SDR such as GNU Radio [12]. GNU Radio is one of the software tools that supports SDR implementation. GNU Radio is open source, allowing it to be used for free, and it supports programming languages like C++ and Python [9].

In [16,17], they've harnessed SDR technology to effectively establish a wireless communication system, enhancing it by incorporating an OFDM system alongside a line code system. This fusion has led to the creation of an effective and robust wireless communication solution.

Moreover, Martianda et al. have also utilized the capabilities of SDR for channel measurements in communication systems with multiple antennas, as elaborated in their publication [18]. The versatility and relevance of SDR across various communication domains are visually depicted in Figure 1, encapsulating the core essence of this transformative technology.

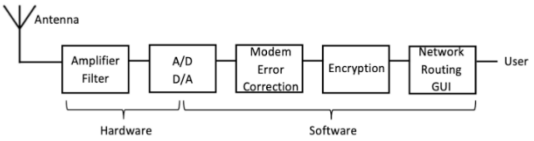
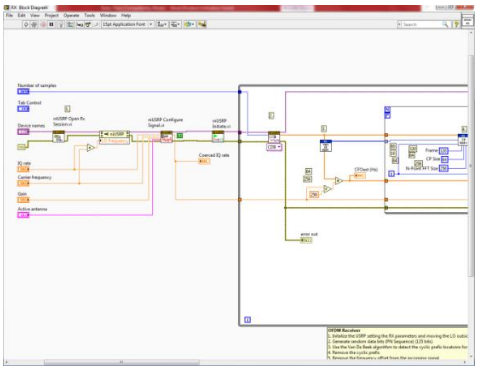


Fig. 1. Concept of Software-Defined Radio

USRP is a hardware device utilized to support real-time GNU Radio, which is a SDR platform. GNU Radio is open source. The USRP consists of a Field Programmable Gate Array (FPGA) for high-speed signal processing, a high-speed digital-to-analog converter (DAC), and I/O ports [19]. There are various companies that manufacture FPGAs, including Xilinx, Altera, Actel, Lattice, and Tabula [9]. FPGAs find extensive use in control systems, signal processing, and coding within telecommunications. One type of USRP is the NI-USRP N2920. The USRP N2920 covers a frequency range of 50MHz to 2.2GHz and employs Gigabit Ethernet for transmitting data to and from the host processor. Programming it involves using Laboratory Virtual Instrument Engineering Workbench (LabVIEW) software. LabVIEW is a system-design platform and development environment for a visual programming language created by National Instruments. The graphical language within LabVIEW is known as "G." This G dataflow language was initially developed by the LabVIEW system. It is widely utilized for tasks such as data acquisition, instrument control, and industrial automation across various operating systems, including macOS, different Unix and Linux versions, and Microsoft Windows. The LabVIEW interface is illustrated in Figure 2.



OFDM Receiver
1. Initialize the USRP setting the RX parameters and moving the LO outside
2. Generate random data bits (PN Sequence) (125 bits)
3. Use the Van De Beek algorithm to detect the cyclic prefix locations for
4. Remove the cyclic prefix
5. Remove the frequency offset from the incoming signal

Fig. 2. LabVIEW software interface

OFDM is one of the currently popular communication technologies. OFDM has found its application in various wireless standards, including Digital Audio Broadcasting (DAB), Digital Video Broadcasting (DVB-T), IEEE 802.11a for Local Area Network (LAN) protocols, and IEEE 802.16a for Metropolitan Area Network (MAN) protocols [20]. OFDM represents an efficient multicarrier bandwidth scheme for digital communication in which the subcarriers overlap, leading to enhanced spectrum efficiency [21].

Orthogonal Frequency Division Multiplexing (OFDM) is a wireless technology that employs a method of dividing carriers into multiple orthogonal subcarriers. By transforming a single-carrier signal according to the Nyquist criterion into a multi-carrier criterion, it generates multiple orthogonal subcarrier signals that overlap in the spectrum. In practical implementation, the Discrete Fourier Transform (DFT) and inverse DFT (IDFT) processes are used to achieve orthogonal signal generation, often performed efficiently using consecutive Fast Fourier Transform (FFT) and Inverse FFT (IFFT). The overlapping frequencies in OFDM serve to conserve frequency usage. Additionally, the OFDM scheme incorporates a guard interval in the time domain known as a cyclic prefix (CP), which reduces inter-symbol interference (ISI) between OFDM symbols [22]. Figure 3 illustrates the transmitter and receiver block diagram within the OFDM system.

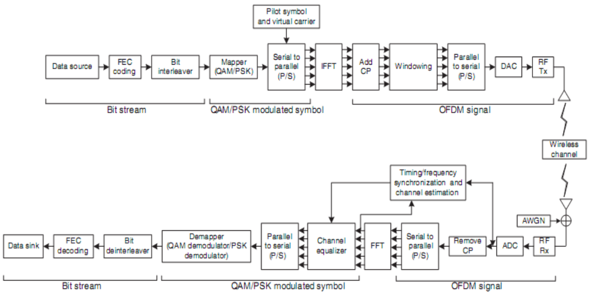


Fig. 3. Scheme of OFDM transmission

In OFDM, each sub-carrier is organized to overlap with one another and possesses orthogonal characteristics. By employing this overlapping technique, channel bandwidth can be conserved. The IFFT and FFT processes are crucial components in OFDM. IFFT serves as the OFDM symbol generator (modulator), while FFT functions as the OFDM symbol decoder (demodulator). The equations for FFT and IFFT can be expressed as (1) and (2):

FFT:

$$x(k) = \sum_{n=0}^{N-1} x(n) \sin\left(\frac{2\pi kn}{N}\right) + j \sum_{n=0}^{N-1} x(n) \cos\left(\frac{2\pi kn}{N}\right) \quad (1)$$

IFFT:

$$x(n) = \sum_{n=0}^{N-1} x(k) \sin\left(\frac{2\pi kn}{N}\right) + j \sum_{n=0}^{N-1} x(k) \cos\left(\frac{2\pi kn}{N}\right) \quad (2)$$

The operating principle of OFDM can be elucidated as follows. First, the sequential data stream intended for transmission is transformed into a parallel format. Consequently, if the original data rate is denoted as R, the data rate in each parallel path becomes R/M, where M represents the number of parallel paths (equivalent to the number of subcarriers). Following this, modulation is applied to each subcarrier, with modulation options including BPSK, QPSK, QAM, and others. Subsequently, these modulated signals are subjected to the Inverse Discrete Fourier Transform (IDFT) process to generate OFDM symbols. The utilization of IDFT ensures the allocation of mutually orthogonal frequencies. Afterward, the OFDM symbols are converted back into a serial format before transmission [22].

The OFDM signal x(t) during the interval $mT_u \leq t \leq (m+1)\ T_u$can be expressed by (3):

$$x(t) = \sum_{k=0}^{N_c-1} x_k\,(t) = \sum_{k=0}^{N_c-1} a_k^{(m)}\, e^{j2\pi k\Delta f t} \quad (3)$$

With: subcarrier modulated to k with frequency $f_k = k.\Delta f$ and $a_k^{(m)}$, and is the modulation symbol applied to subcarrier k during the OFDM interval m [10]. The term OFDM relates to the fact that two OFDM subcarriers $x_{k1}(t)$ and $x_{k2}(t)$and are orthogonal, as indicated by (4).

$$\int_{mT_u}^{(m+1)T_u} x_{k1}(t) x_{k2}^{*}\,(t) dt = 0 \qquad \text{for } k_1 \neq k_2 \quad (4)$$

A guard time, often referred to as a cyclic prefix (CP) or interval, is inserted between the sequence of OFDM symbols. Its primary purpose is to mitigate Inter-Symbol Interference (ISI). As illustrated in Figure 4, the CP is essentially a duplicate of the latter portion of the OFDM symbol that precedes the symbol intended for transmission. By including this cyclic prefix, the OFDM signal becomes less susceptible to the effects of time dispersion, provided that the time dispersion does not exceed the length of the cyclic prefix [22]. The components of dispersion are discarded before information retrieval.

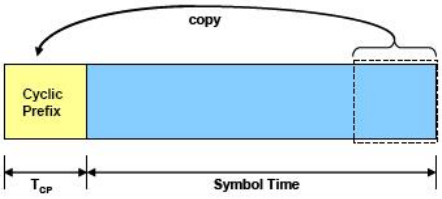


Fig. 4. Adding cyclic prefix

p-ISSN 2301-7988, e-ISSN 2581-0588
DOI : 10.32736/sisfokom.v13i2.2024, Copyright ©2024
Submitted : November 29, 2023, Revised : April 16, 2024 , Accepted : April 25, 2024, Published : June 5, 2024

The OFDM symbol is transformed from the symbol time T_u to $T_u + T_{CP}$, where T_{CP} is the length of the CP. If correlation at the receiver is still performed within the interval $T_u = 1/\Delta f$, orthogonality will be preserved as it doesn't overlap with other symbols.

There are two notable advantages to incorporating a cyclic prefix (CP) into an OFDM symbol. Firstly, the CP acts as a protective buffer between two successive symbols, effectively preventing ISI by providing a guard space. Secondly, the CP transforms the linear convolution with the channel impulse response into cyclic convolution. In the time domain, this cyclic convolution simplifies to scalar multiplication in the frequency domain. This property helps maintain subcarrier orthogonality and mitigates Inter-Carrier Interference (ICI). [22].

In an OFDM system, it's possible to calculate the throughput from the received signal on the receiver side. The equation for calculating throughput can be represented as (5).

$$Throughput = \frac{N}{T} = \frac{D \times m \times r}{T} \tag{5}$$

Where:

N = Number of bits per OFDM symbol

T = Duration of an OFDM symbol

D = Data of the OFDM symbol

m = Modulation bits (m = 1 for BPSK; m = 2 for 4QAM; for 16QAM, m = 4)

r = Ratio of CP length to symbol length

Convolutional coding is a type of channel coding technique. Its encoding involves registers containing shift registers and XOR gates. Convolutional codes are characterized by two important parameters namely the code rate (R) and the memory length (K). The code rate represents the ratio between input information bits and the resulting encoded output bits. It takes the form of an equation like in (6).

$$R = \frac{K}{n} \tag{6}$$

Where:

R = code rate of convolutional code

K = number of the convolutional code input bits

N = number of the convolutional code output bits

Memory length (K) refers to the memory with the current input bit in the Convolutional Code. The memory length can be defined using an equation like in (7).

$$K = M + 1 \tag{7}$$

Where,

K = Memory length

m = Memory

The generator polynomial is a key component of a convolutional code. It is defined as the impulse response to the memory of the encoder's shift register. In simpler terms, it describes the specific part of the shift register's memory that is connected to each adder within the encoding process [23]. Additionally, the block diagram of a convolutional code with a code rate of 1/2 and a memory length (K) of 3 is depicted in Figure 5. This diagram illustrates how the shift register, adders, and generator polynomial work together to encode data in the convolutional coding process.

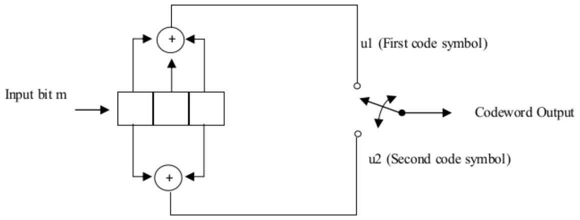


Fig. 5. Convolutional Code block diagram (rate ½, K = 3) [17]

The state of a convolution code (n, 1, m) is defined as the contents of its first m - 1 registers. Thus, a Convolutional encoder can be represented by a finite state machine with (m-1) states. By knowing the state of the encoder at the next time step, along with the output it will produce, we can determine the state transition. The initial state, or state zero, is when all (m-1) of the first registers of the encoder contain the value 0. The transition from one state to the next is determined by the input bit (0 or 1) that will be given at that next time step. A binary Convolutional code has 2M-1 possible states. The state diagram representation for the Convolutional code can be seen in Figure 6. In this specific case where m=3, there are 4 possible states for the encoder: a = 00, b = 10, c = 01, d = 11 [23].

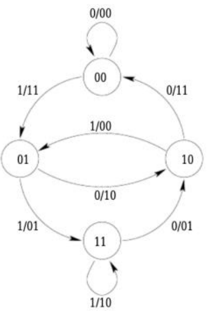


Fig. 6. Diagram state [20]

The encoding process with Convolutional codes can be depicted using a trellis diagram. This Convolutional code has four states, which are placed along the vertical axis. The transitions between these states are represented by vertical and diagonal lines that crisscross (resembling a grid or lattice) and move to the right as time progresses.

The Viterbi algorithm implemented with Maximum Likelihood (ML) in hard decision. A convolutional encoder encodes original information, for example, a sequence like (11010100), and generates an encoded output with a code rate of 1/2, resulting in a sequence like (11, 01, 01, 00, 10, 00, 10, 11). This encoded output is then transmitted through a channel that is affected by noise. At the receiver's end, let's assume we obtain the received sequence (10, 01, 01, 01, 10, 00, 10, 11). By comparing the output sequence from the convolutional encoder and the received sequence at the Viterbi decoder, differences can be observed due to the noise introduced in the channel.

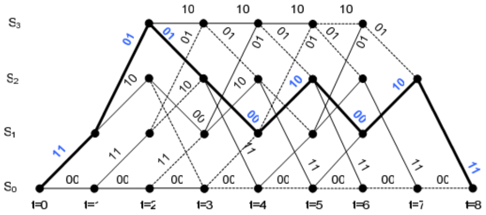


Fig. 7. Trellis Diagram [23]

The traceback procedure in the Viterbi decoder is depicted in the trellis diagram shown in Figure 7. The traceback process is based on the idea that each branch in the encoder is associated with a specific input bit. For instance, when tracing the path from state S2 at time t = 7 to state S0 at time t = 8, this corresponds to the input bit '0' that was processed by the encoder at that particular point in time [23]. By integrating Convolutional Code and Viterbi Decoder blocks into OFDM, the aim is to reduce the Bit Error Rate (BER) value [24]-[26].

B. System Design

This section explains the system design for the proposed reliable communication system between the TPMS and the server. In this proposal, we use computers and USRP devices on both the TPMS and server sides. The device configuration is illustrated in Figure 8. While the specification of the device is showed in Table 1.

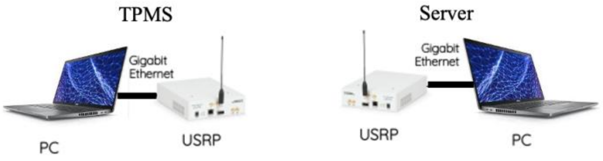


Fig. 8. Hardware configuration

TABLE I. SPECIFICATION OF DEVICES

No	Device	Specification
1	Computer (Dell Precision T1650)	Processor: Intel® Xeon® Quad Core 3.2 GHz
		RAM: 4 GB
		Processor: Intel® Xeon® Quad Core 3.2 GHz
2	USRP (NI USRP 2920)	Frequency: 50 MHz ~ 2.2 GHz
		Output Power: 17dBm ~ 20dBm
		Gain: 0dB ~ 31dB
3	Antenna (VERT900)	Frequency: 824 ~ 960MHz
		Gain: 3dBi

The diagram block system is displayed in Figure 9. On the transmitter side, the initial step involves configuring the IP address of the USRP and establishing a session with it. Next step is configuring the IQ rate and carrier frequency settings. The following steps involve reading a text file containing data and converting the text data into a sequence of bits.

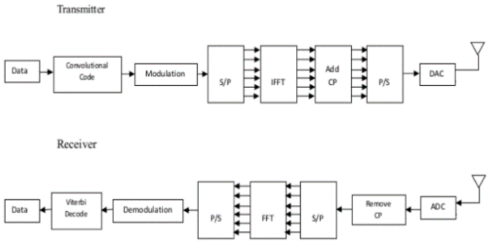


Fig. 9. Diagram block system

This 17-byte file is then transformed into 136 data bits. Subsequently, these data bits are encoded using Convolutional Code, resulting in the generation of codewords. The program for Convolutional Code is shown in Figure 10.

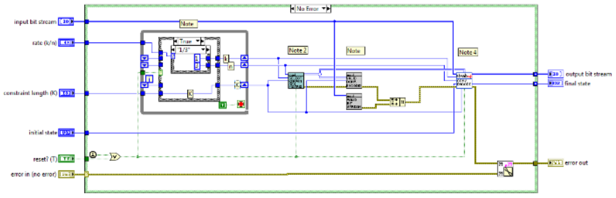


Fig. 10. Convolutional code program

Afterward, the codeword is modulated into symbols using the Bits to Symbols Mapping function and adjusted according to the type of modulation being used. This function requires a symbol map for each modulation. Several types of modulations are employed, including BPSK, QPSK, 4QAM, and 16QAM. For BPSK modulation, PSK is selected. The symbol map represents the values of symbols in the form of complex numbers and is already normalized. For BPSK modulation, the symbol map uses 1 + 0j and -1 + 0j. For M-QAM modulation, the value α is utilized as described in (8).

$$\alpha_{M-QAM}= [\pm(2m-1)\pm(2m-1)] \qquad (8)$$

Where $m \in \left\{1,\dots,\frac{\sqrt{M}}{2}\right\}$. For example, in 16QAM (M = 16), the size of the modulation α can be determined as expressed in (9).

$$\alpha_{16QAM}=\begin{Bmatrix}\pm1+\pm1j & \pm1+\pm3j\\ \pm3+\pm3j & \pm3+\pm1j\end{Bmatrix} \qquad (9)$$

The average energy of the constellation is expressed in (10).

$$E_{M-QAM}=\frac{2}{3}\,(M-1) \qquad (10)$$

The average energy for 16-QAM is 2/3(16-1) = 10. Therefore, to normalize the average energy to one, a scaling factor of $1/\sqrt{10}$ is used. The program for modulation is illustrated in Figure 11.

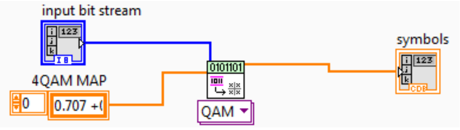


Fig. 11. Modulation program

p-ISSN 2301-7988, e-ISSN 2581-0588
DOI : 10.32736/sisfokom.v13i2.2024, Copyright ©2024
Submitted : November 29, 2023, Revised : April 16, 2024 , Accepted : April 25, 2024, Published : June 5, 2024

The signal transmission process involves using the function niUSRP Write Tx Data (poly). Once the signal is sent, the USRP session is closed using the function niUSRP Close Session. The signal transmission process is illustrated in Figure 12.

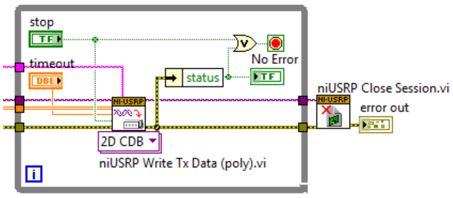


Fig. 12. Signal transmission program

The received analog signal is processed by another USRP unit. Initially, the analog signal undergoes conversion into a digital signal. Afterward, the cyclic prefix is stripped from the signal, and it is parallelized. The signal is subsequently transformed from the time domain to the frequency domain. Zero-padding is removed from the symbols, and pilot symbols are extracted to serve as a reference for the channel estimator. Figure 13 shows the channel estimator program.

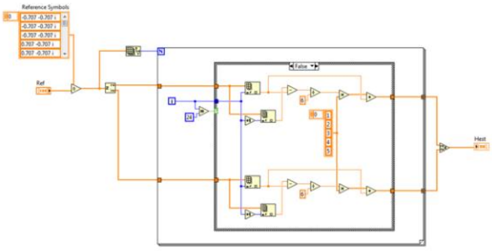


Fig. 13. Channel estimator program

The next step is equalization. Equalization is performed to improve the received symbols, thereby obtaining better data. Equalization in this thesis employs the zero-forcing method. Zero-forcing is an equalization method that involves multiplying channel estimates with data. The zero-forcing equalization program is illustrated in Figure 14.

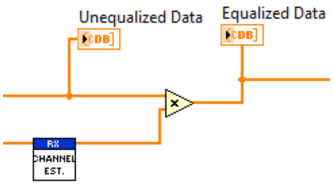


Fig. 14. Zero-forcing equalization program

Following this, the signal is demodulated into bits using program in Figure 15. After that, the signal is decoded using the Viterbi decoder. Figure 16 shows the Viterbi decoder program.

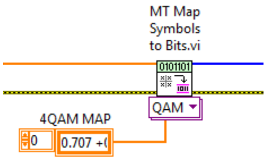


Fig. 15. Demodulation program

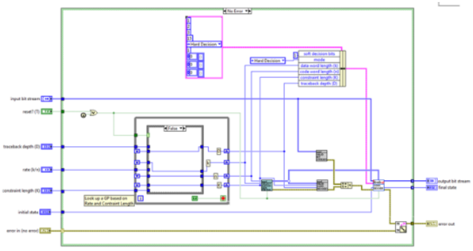


Fig. 16. Viterbi decoder program

The resulting data is then converted back into a text file and compared to the original data that was transmitted. The system parameters are outlined in Table 2.

TABLE II. SYSTEM PARAMETERS

	Parameters	**Value**
Transmitter	Symbol OFDM	136bits
	Pilot Sequence	34 symbols
	Subcarriers	256
	FFT Size	256
	Guard Intervals	86bits
	Carrier Frequency	910MHz
	Cyclic Prefix Ratio	1/4
Receiver	Equalizer	Zero forcing

III. RESULTS AND DISCUSSION

This chapter discusses the experiment results. We conducted the several experiments to evaluate the system performance.

A. Constellation Diagram Measurement

First, we measure the constellation performance of the system. In this measurement, we use single carrier and OFDM communication. The measurements utilize an antenna on the transmitter side and an antenna on the receiver side. These measurements employ 915MHz carrier frequency and a rate of 250k symbols/s. The amount of data is 8184 samples, with 1 symbol consisting of 8 samples. BPSK, 4QAM and 16QAM modulation are used in these measurements. The constellation diagrams on the receiver side are illustrated in Figure 17. Based on the constellation diagram, the system remains stable up to 16QAM modulation.

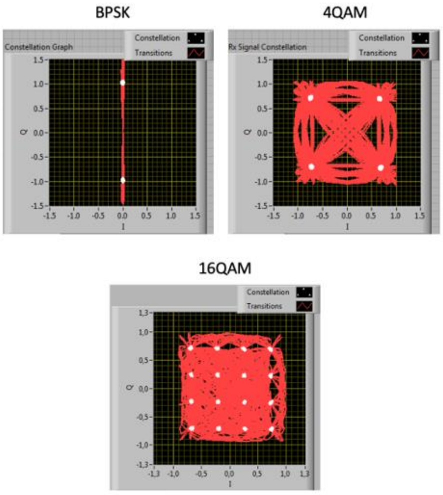


Fig. 17. Constellation diagram

B. *Throughput Measurement*

The next measurement is the throughput measurement on a single carrier and OFDM system with BPSK, 4QAM, and 16QAM modulations. The throughput results are depicted in Figure 18. Based on the results, the best performance is OFDM with 16QAM modulation.

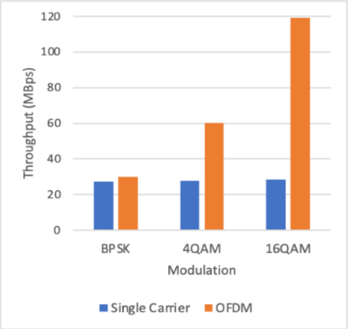


Fig. 18. Throughput measurement for single carrier and OFDM system

The calculation of throughput is based on equation 5 and is compared with the measurement results. As a result, the data from the calculated and measured throughput of OFDM with BPSK, 4QAM, and 16QAM modulations is presented in Table 3. Based on the results, the error between theory and measurement is still relatively small.

TABLE III. COMPARISON OF THROUGHPUT: THEORETICAL CALCULATION VS. MEASUREMENT IN OFDM SYSTEM

Modulation	**Calculated Throughput (MBps)**	**Measured Throughput (MBps)**	**Error (%)**
BPSK	30.7	30.09	1.99
4QAM	61.61	60.27	2.17
16QAM	122.81	119.19	2.95

C. *Bit Error Measurement*

The next measurement involves transmitting data using OFDM with Convolutional Code channel coding. The data sent is a text file with a size of 104 bits. The data is transmitted through 4QAM modulation scheme. Convolutional Codes (CV) with rates of 1/2, 1/3, 1/4, 2/3, and 3/4 are employed. The results are displayed in Table 4.

TABLE IV. BIT ERROR RESULTS

Type of Code	**Error (bits)**
Uncoded	6
CV with 1/2 code rate	2
CV with 1/3 code rate	2
CV with 1/4 code rate	2
CV with 2/3 code rate	0
CV with 3/4 code rate	0

In OFDM using CV with code rates of 1/2, 1/3, and 1/4, there were only 2 bits of error, reducing the bit error rate by 3.85% compared to not using channel coding. Then, when using code rates of 2/3 and 3/4, all characters were received well, resulting in the bit error rate reduction of 5.77%. The performance of code rates 2/3 and 3/4 is superior to that of 1/2, 1/3, and 1/4.

IV. CONCLUSION

The Tire Pressure Monitoring System (TPMS) has become a crucial component of modern vehicles, significantly contributing to road safety and improving the driving experience. Traditionally, TPMS systems have depended on specific hardware components to gather and send tire pressure data to the vehicle's onboard computer. In this paper, we developed an advanced communication system between the TPMS and the server based on the SDR. This communication system utilizes OFDM and Convolutional Code. OFDM is employed to enhance throughput, while Convolutional Code is used to reduce received data errors. Experimental results indicate that using the OFDM system with 16QAM modulation achieves a maximum throughput of 119.19MBps. The bit error rate without coding is 5.77%. However, the use of Convolutional Code with a code rate of 1/2 can reduce the received bit error rate to 3.85%. Then, when using code rates of 2/3 and 3/4, all characters were received well. Thus, this system enhances TPMS communication reliability with the server while maintaining high throughput. For further research, we plan to integrate a security protocol into the communication system between TPMS and server, ensuring that transmitted data remains resistant to unauthorized access or hacking attempts.

ACKNOWLEDGMENT

We would like to thank Telkom University, Indonesia for supporting funding to our research.

REFERENCES

[1] S. Formentin, L. Onesto, T. Colombo, A. Pozzato, S. M. Savaresi, “H-TPMS: A Hybrid Tire Pressure Monitoring System for Road Vehicles,” Mechatronics, vol. 74, 2021. https://doi.org/10.1016/j.mechatronics.2021.102492

[2] A. Abbi and T. Ramakrishnaiah, “Tyre Pressure Monitoring System,” IOP

Conference Series: Materials Science and Engineering, vol. 1024, pp. 1-7, 2021.

https://doi.org/10.1088/1757-899X/1042/1/012024

[3] G. Phade, A. Kulkarni, O. Vaidya, and S. Gandhe, "Wireless Communication Based Effective Tire Pressure Monitoring System," International Journal of Advanced Research in Engineering and Technology (IJARET), vol. 11, issue 11, pp.1282-1291, Nov. 2020. https://doi.org/10.34218/IJARET.11.11.2020.116.

[4] S. Hwang, S. Kim, J. Pyun, and G. Kwon, "Low-power TPMS Data Transmission Technique Based on Optimal Tire Condition," The Second International Conference on Advances in Vehicular Systems, Technologies and Applications, France, pp. 29-34, 2013.

[5] S. Gryś, "An Experimental Test Bench for the Tire Pressure Monitoring System – Discussion of Measurement and Communication Issues," International Journal of Electronics and Telecommunications, vol. 65, no. 1, pp. 51-56, Jan. 2019. https://doi.org/10.24425/123565.

[6] A. Digulescu, A. Sârbu, D. Stanescu, D. Nastasiu, C. Despina-Stoian, C. Ioana, A. Mansour, "Detection of OFDM Modulations Based on The Characterization in The Phase Diagram Domain," Frontiers in Signal Processing, vol. 3, pp. 1-12, Aug. 2023.

https://doi.org/10.3389/frsip.2023.1197590

[7] S. A. Matin and L. B. Milstein, "OFDM System Performance, Variability and Optimality with Design Imperfections and Channel Impediments," IEEE Transactions on Vehicular Technology, vol. 70, pp. 381-397, Dec. 2021.

https://doi.org/10.1109/TVT.2020.3044665

[8] M. M. Gulo, I. G. P. Astawa, Y. Moegiharto, H. Briantoro, "The Joint Channel Coding and Pre-Distortion Technique on The USRP-Based MIMO-OFDM System, RESTI, vol. 7, no. 4, pp. 930-939, Aug, 2023.

https://doi.org/10.29207/resti.v7i4.5093

[9] S. Mori, K. Mizutani and H. Harada, "Software-Defined Radio-Based 5G Physical Layer Experimental Platform for Highly Mobile Environments," IEEE Open Journal of Vehicular Technology, vol. 4, pp. 230-240, Jan. 2023.

https://doi.org/10.1109/OJVT.2023.3237390

[10] S. Gökceli et al., "SDR Prototype for Clipped and Fast-Convolution Filtered OFDM for 5G New Radio Uplink," IEEE Access, vol. 8, pp. 89946-89963, May 2020.

https://doi.org/10.1109/ACCESS.2020.2993871

[11] A. B. Muturatnam, N. V. Sridharan, A. P. Sreelatha, and S. Vaithiyanathan, "Enhanced Tyre Pressure Monitoring System for Nitrogen Filled Tyres Using Deep Learning," Machines, vol. 11, no. 4, pp. 1-7, Mar. 2023.

https://doi.org/10.3390/machines11040434

[12] A. Silva, J. R. Sánchez, G. E. Granados, J. C. Tudon-Martinez, J. J. Lozoya-Santos, "Comparative Analysis in Indirect Tire Pressure Monitoring Systems in vehicles," IFAC-Papers Online, vol. 52, no. 5, pp. 54-59, 2019.

https://doi.org/10.1016/j.ifacol.2019.09.009

[13] H. Fechtner, U. Spaeth and B. Schmuelling, "Smart Tire Pressure Monitoring System with Piezoresistive Pressure Sensors and Bluetooth 5," 2019 IEEE Conference on Wireless Sensors (ICWiSe), Pulau Pinang, Malaysia, 2019, pp. 18-23, https://doi.org/10.1109/ICWISE47561.2019.8971835

[14] A. Aboltins and N. Tihomorskis, "Software-Defined Radio Implementation and Performance Evaluation of Frequency-Modulated Antipodal Chaos Shift Keying Communication System," Electronics, vol. 12, no. 5, pp. 1240-1260, 2023.

https://doi.org/10.3390/electronics12051240

[15] KK. M. Reddy, S. J. Darak and M. D. Praveen, "Novel Framework for Enabling Hardware Acceleration in GNU Radio," 2020 IEEE International Symposium on Circuits and Systems (ISCAS), Seville, Spain, 2020, pp. 1-5.

https://doi.org/ 10.1109/ISCAS45731.2020.9181226.

[16] H. Briantoro, I. G. P. Astawa, and A. Sudarsono "An Implementation of Error Minimization Data Transmission in OFDM Using Modified Convolutional Code, EMITTER International Journal of Engineering Technology, vol. 3, no. 2, pp. 43-59, Jul, 2016.

https://doi.org/10.24003/emitter.v3i2.44

[17] H. Briantoro, I. G. P. Astawa, and A. Sudarsono, "Implementation of Digital Modulation With Single Carrier And Multi Carrier OFDM Schemes Using USRP," 2014 International Electronic Symposium, Surabaya, Indonesia, pp. 1-6.

[18] M. E. Anggraeni, P. Handayani and G. Hendrantoro, "Double-Directional Outdoor MIMO Channel Measurement at 2.4 GHz Using SDR," 2016 International Seminar on Intelligent Technology and Its Applications (ISITIA), Lombok, Indonesia, 2016, pp. 255-260.

https://doi.org/10.1109/ISITIA.2016.7828667

[19] E. A. Thompson, N. Clem, I. Renninger, and T. Loos, "Software-defined GPS Receiver on USRP-Platform," Journal of Network and Computer Applications, vol. 35, no. 4, pp.1352-1360, 2012.

https://doi.org/10.1016/j.jnca.2012.01.020.

[20] R. Sharma, G. Singh, and R. Agnihotri, "Comparison of Performance Analysis of 802.11a, 802.11b and 802.11g Standard," International Journal on Computer Science and Engineering, vol. 2, no. 6, pp. 2042-2046, 2010.

[21] S. P. Girija, et al., Fractional Weighted ZF Equalizer: A Novel Approach for Channel Equalization in MIMO-OFDM System Under Impulse Noise Environment, Communications in Science and Technology Journa, vol. 6, no.1, pp. 1-10, 2021.

https://doi.org/10.21924/cst.6.1.2021.224

[22] Y. S. Cho, J. Kim, W. Y. Yang, and C.-G. Kang, "MIMO-OFDM Wireless Communications With MATLAB," Wiley IEEE Press, Aug. 2010.

https://doi.org/10.1002/9780470825631

[23] J. Tuominen and J. Plosila, "Asynchronous Viterbi Decoder in Action Systems," Turku Centre for Computer Science, 2005.

[24] Y. Jiang, "Analysis of Bit Error Rate Between BCH Code and Convolutional Code in Picture Transmission," 2022 3rd International Conference on Electronic Communication and Artificial Intelligence (IWECAI), Zhuhai, China, 2022, pp. 77-80.

https://doi.org/10.1109/IWECAI55315.2022.00023

[25] M. G. Rashed, M. H. Kabir, M. S. Reza, M. M. Islam, R. A. Shams, S. Masum, and S. E. Ullah, Transmission of Voice Signal: BER Performance Analysis of Different FEC Schemes Based OFDM System Over Various Channels, International Journal of Advanced Science and Technology, vol. 34, pp. 89-100, Sept. 2011.

https://doi.org/10.48550/arXiv.1207.3875

[26] H. F. Mohammed and G. A. Al-Rubaye, "BER performance of convolutional coded OFDM system in different fading channels scenarios with exact and simplified LLR," Webology Journal, vol. 19, no. 1, pp. 6300-6321, 2022

Submitted : November 29, 2023, Revised : April 16, 2024 , Accepted : April 25, 2024, Published : June 5, 2024